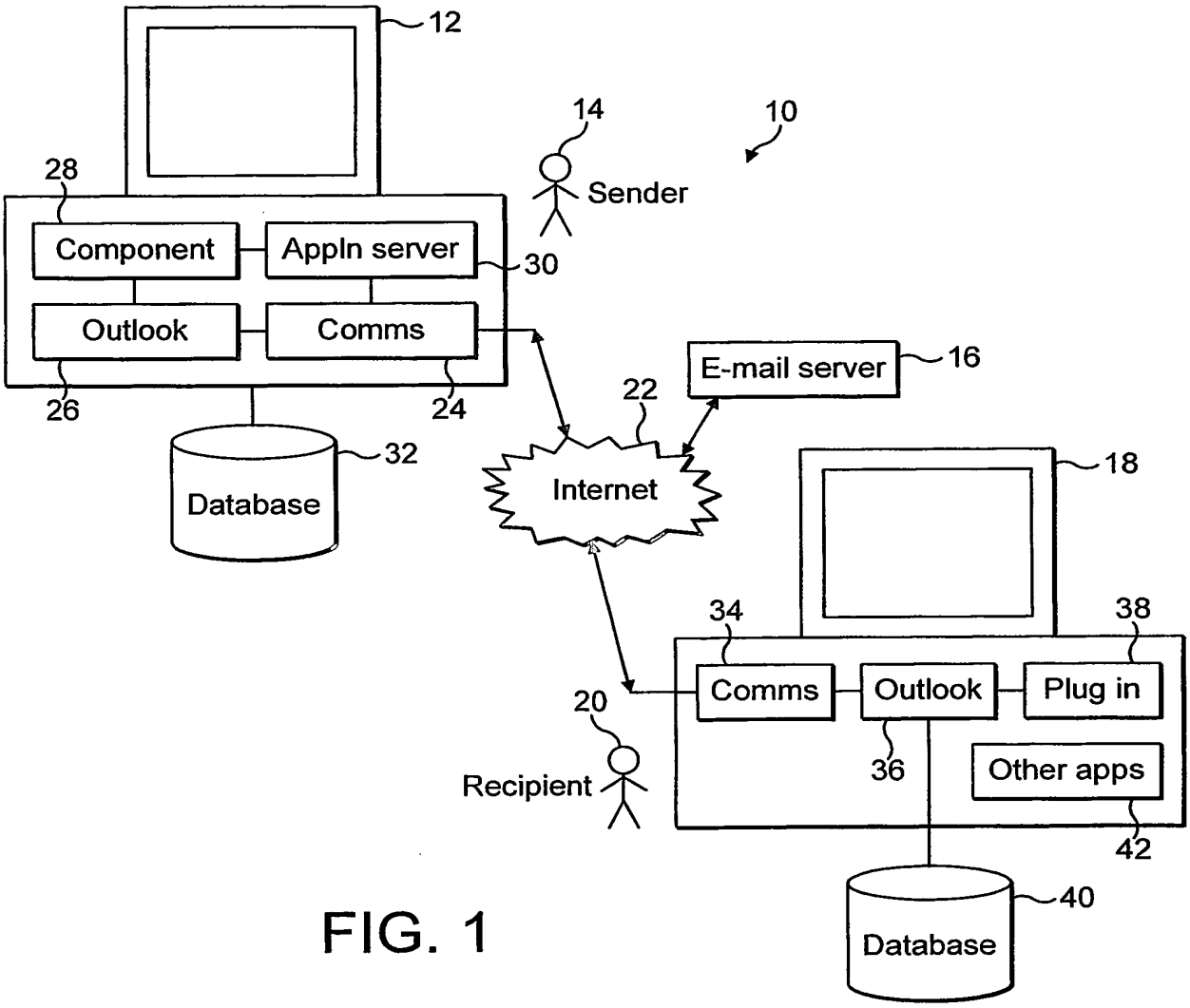
12
14
10
Sender
28
Component
Appln server
30
Outlook
Comms
26
24
E-mail server
16
22
Database
32
Internet
18
34
38
Comms
Outlook
Plug in
20
Recipient
36
Other apps
42
40
Database

FIG. 1

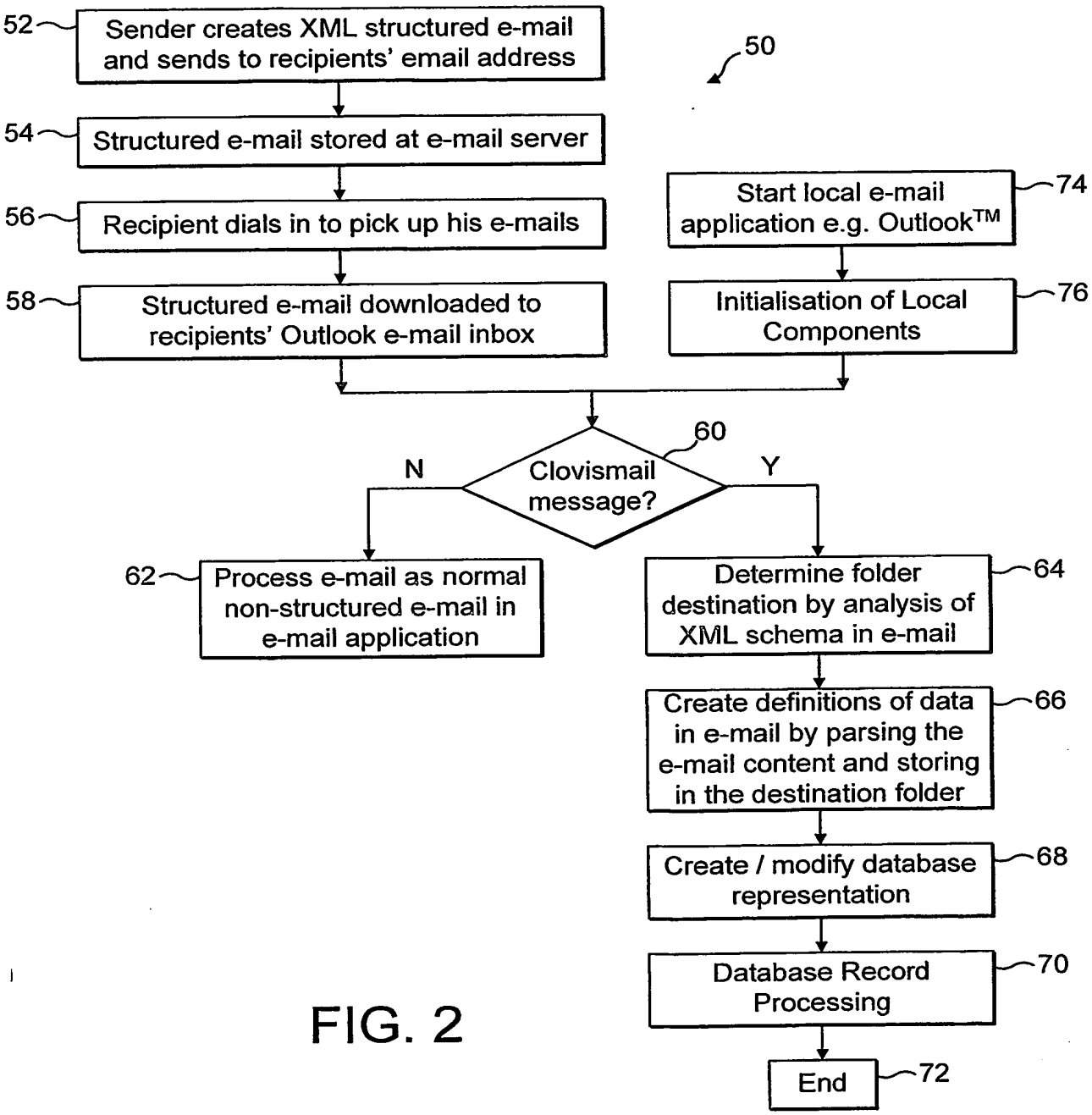
52
Sender creates XML structured e-mail and sends to recipients' email address
50
54
Structured e-mail stored at e-mail server
56
Recipient dials in to pick up his e-mails
58
Structured e-mail downloaded to recipients' Outlook e-mail inbox
Start local e-mail application e.g. Outlook™
74
Initialisation of Local Components
76
60
Clovismail message?
N
Y
62
Process e-mail as normal non-structured e-mail in e-mail application
Determine folder destination by analysis of XML schema in e-mail
64
Create definitions of data in e-mail by parsing the e-mail content and storing in the destination folder
66
Create / modify database representation
68
Database Record Processing
70
End
72

FIG. 2

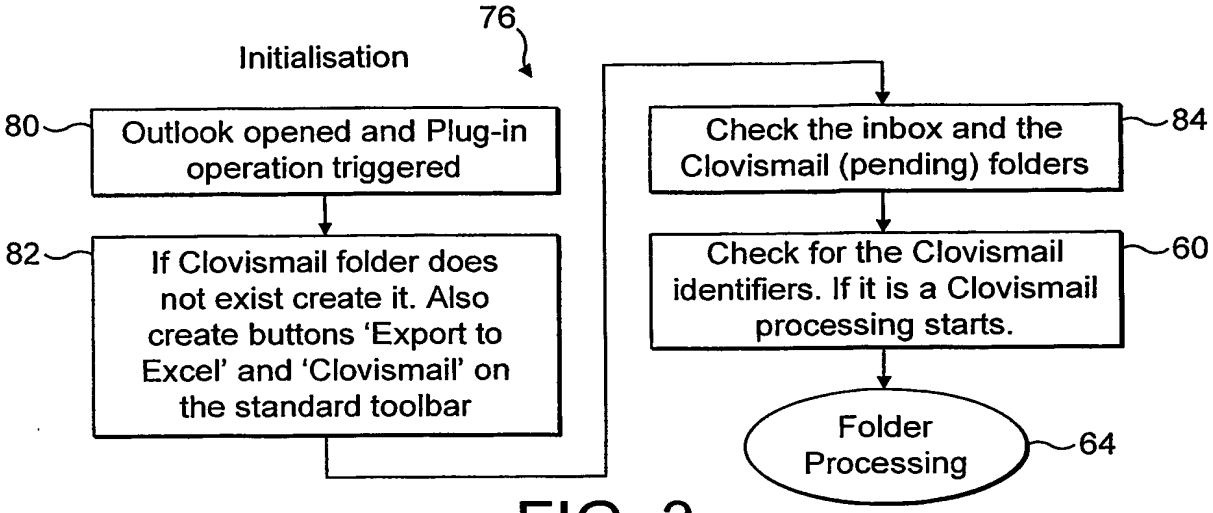
76
Initialisation
80
Outlook opened and Plug-in operation triggered
82
If Clovismail folder does not exist create it. Also create buttons 'Export to Excel' and 'Clovismail' on the standard toolbar
84
Check the inbox and the Clovismail (pending) folders
60
Check for the Clovismail identifiers. If it is a Clovismail processing starts.
Folder Processing
64

FIG. 3

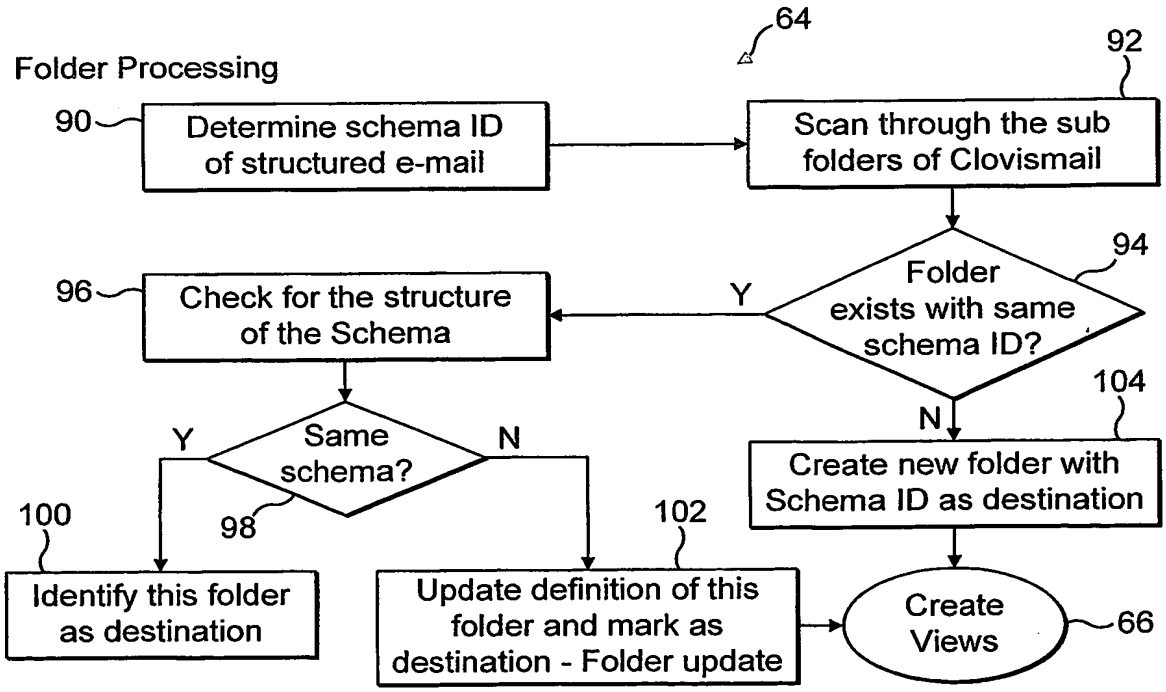
64
Folder Processing
90
Determine schema ID of structured e-mail
92
Scan through the sub folders of Clovismail
94
Folder exists with same schema ID?
Y
96
Check for the structure of the Schema
N
104
Create new folder with Schema ID as destination
Y
Same schema?
N
98
100
Identify this folder as destination
102
Update definition of this folder and mark as destination - Folder update
Create Views
66

FIG. 4

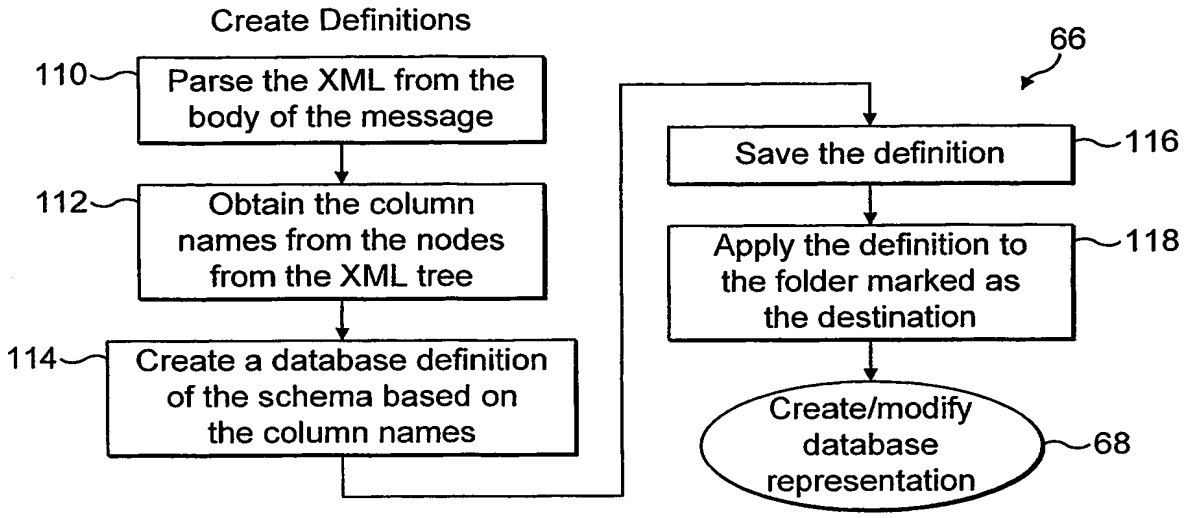
Create Definitions
110
Parse the XML from the body of the message
112
Obtain the column names from the nodes from the XML tree
114
Create a database definition of the schema based on the column names
66
Save the definition
116
Apply the definition to the folder marked as the destination
118
Create/modify database representation
68

FIG. 5

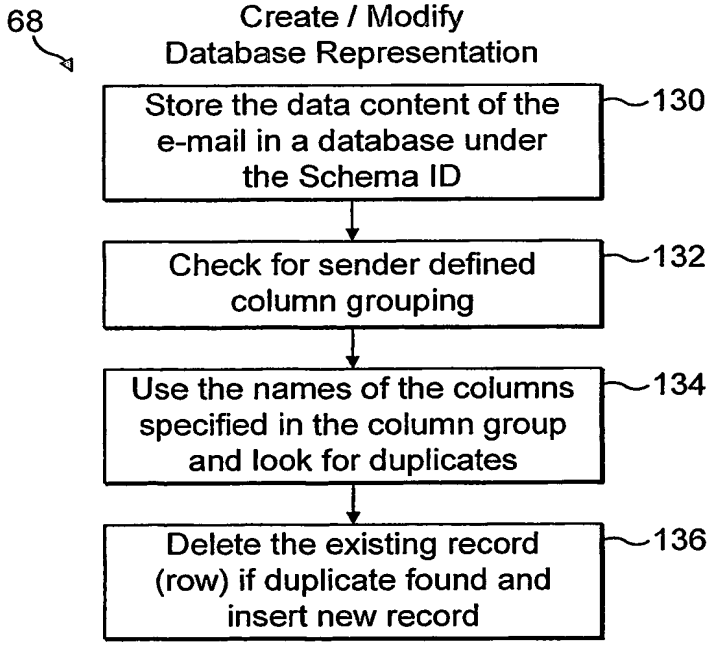
68
Create / Modify Database Representation
Store the data content of the e-mail in a database under the Schema ID
130
Check for sender defined column grouping
132
Use the names of the columns specified in the column group and look for duplicates
134
Delete the existing record (row) if duplicate found and insert new record
136

FIG. 6

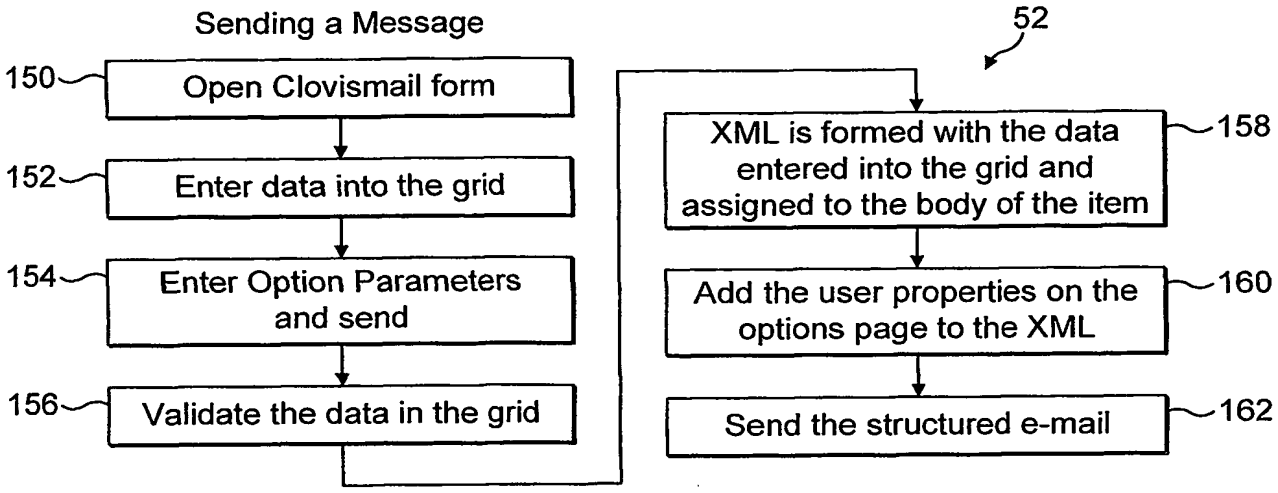
Sending a Message
150
Open Clovismail form
152
Enter data into the grid
154
Enter Option Parameters and send
156
Validate the data in the grid
52
XML is formed with the data entered into the grid and assigned to the body of the item
158
Add the user properties on the options page to the XML
160
Send the structured e-mail
162

FIG. 7

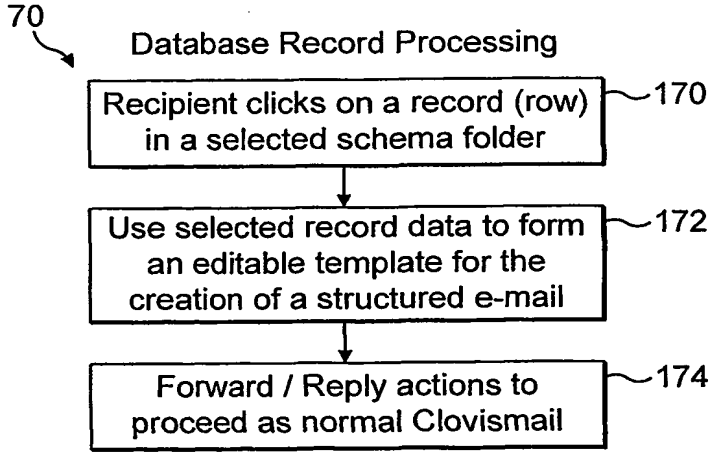
70
Database Record Processing
Recipient clicks on a record (row) in a selected schema folder
170
Use selected record data to form an editable template for the creation of a structured e-mail
172
Forward / Reply actions to proceed as normal Clovismail
174

FIG. 8

BEST AVAILABLE COPY

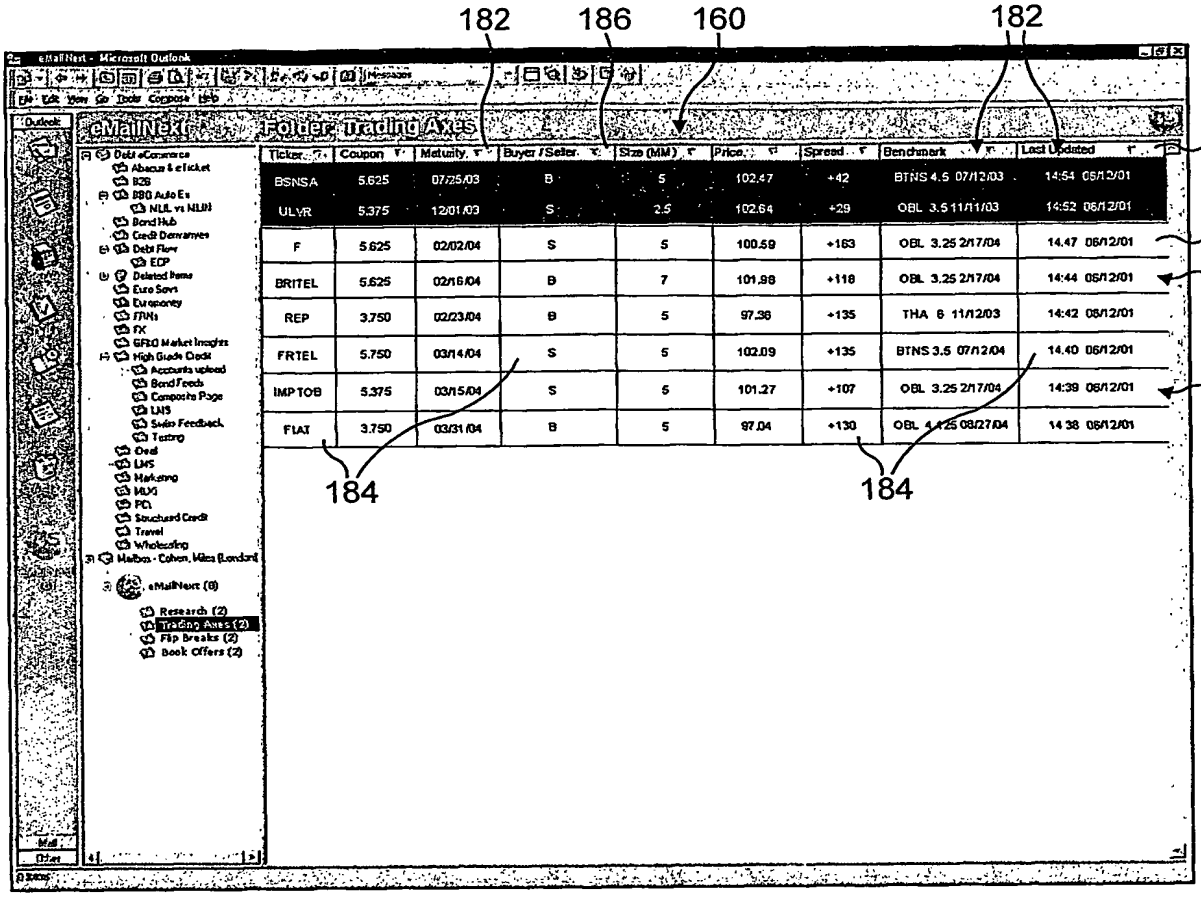

FIG. 9

```
<SCHEMA>                                                    190
192 ->      <FIELD1 Type ="text">Ticker</FIELD1>
            <FIELD2 Type ="text">Coupon</FIELD2>
etc...
                                                            194
            <FIELD8 Type ="text">Last Updated</FIELD8>
            <TABLE KEY>
                  <Ticker>
                  <Coupon>          196
                  <Maturity>
etc...
            </TABLE KEY>                                    202
             <SCHEMA ID = "Trading Axes">
</SCHEMA>

<DATA>                                  198
      <RECORD1>
            <Ticker>BSNSA</Ticker>                          200
            <Coupon>5.625</Coupon>   etc....
      </RECORD1>
                                                            200
      <RECORD2>  etc.....

</DATA>
```

FIG. 10

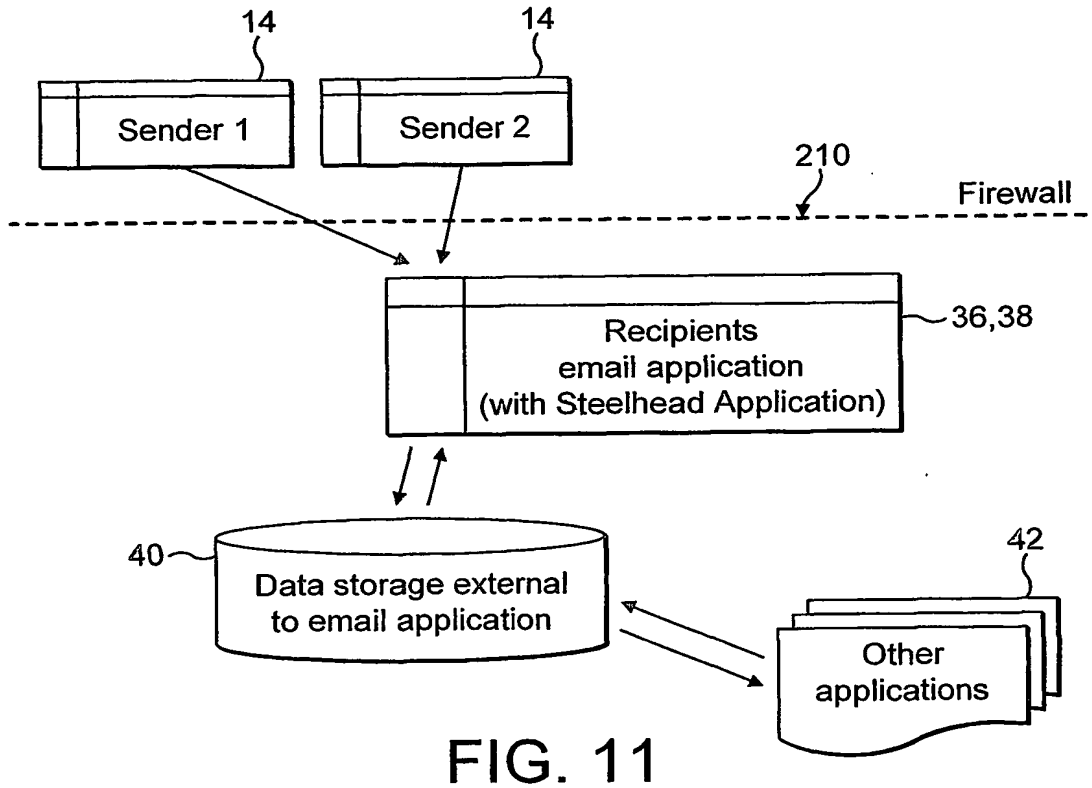
14
14
Sender 1
Sender 2
210
Firewall
Recipients
email application
(with Steelhead Application)
36,38
40
Data storage external
to email application
42
Other
applications

FIG. 11

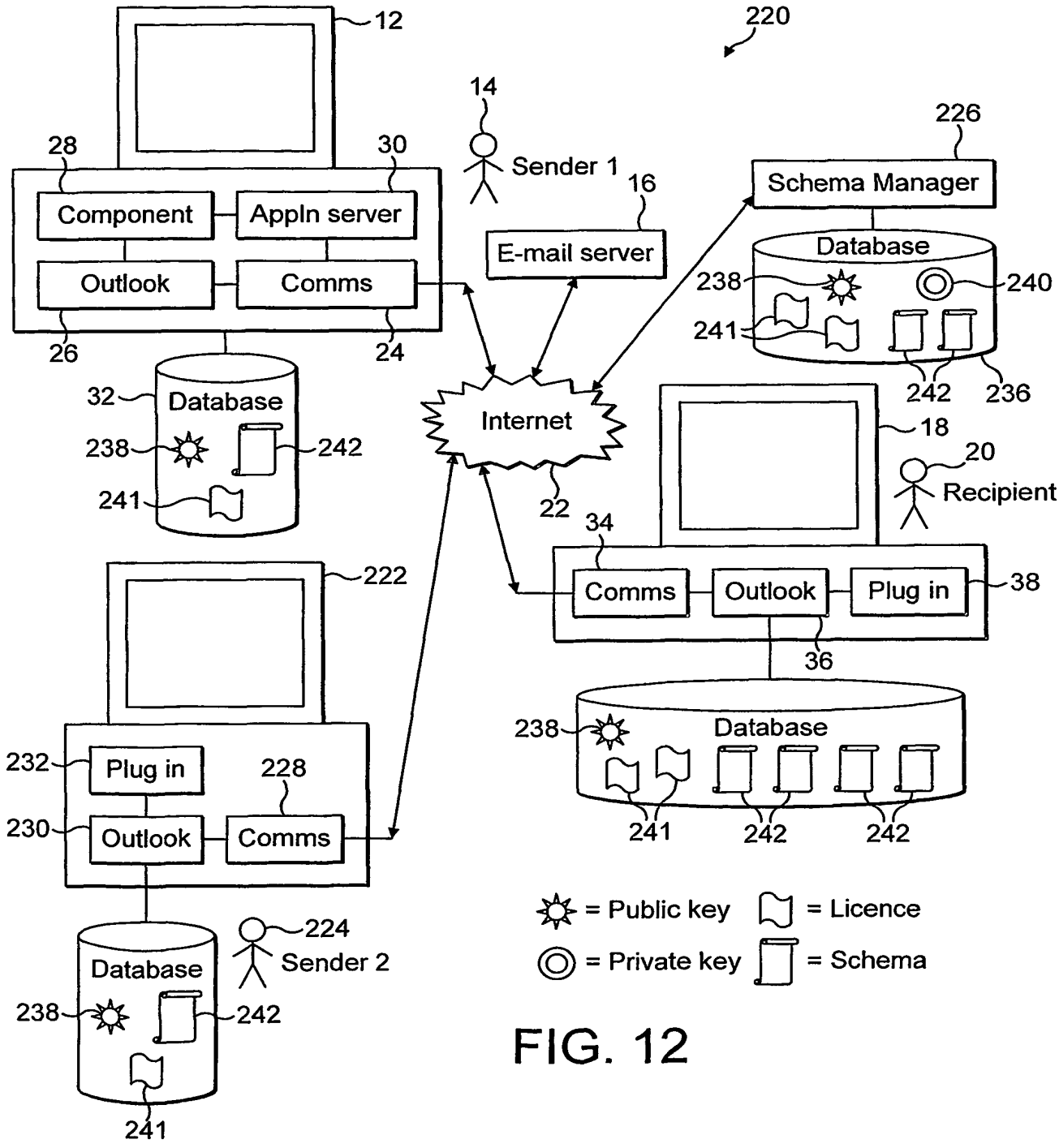
12
220
14
Sender 1
16
226
Schema Manager
28
30
Component
Appln server
E-mail server
Database
Outlook
Comms
238
240
241
26
24
242
236
32
Database
Internet
18
238
242
20
241
Recipient
22
34
222
Comms
Outlook
Plug in
38
36
238
Database
232
Plug in
228
230
Outlook
Comms
241
242
242
= Public key
= Licence
224
Sender 2
= Private key
= Schema
Database
238
242
241

FIG. 12

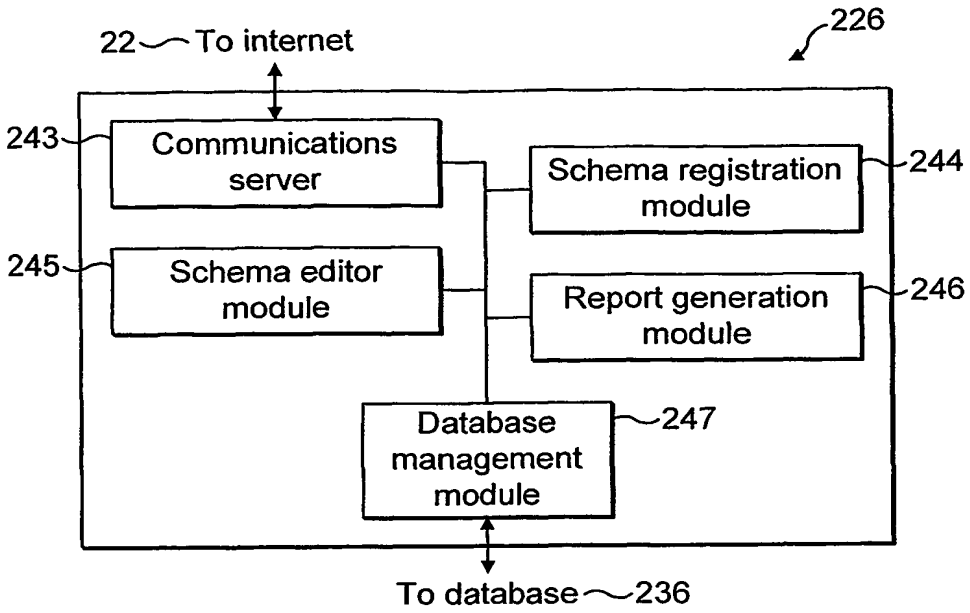
22
To internet
226
243
Communications server
244
Schema registration module
245
Schema editor module
246
Report generation module
247
Database management module
To database
236

FIG. 13

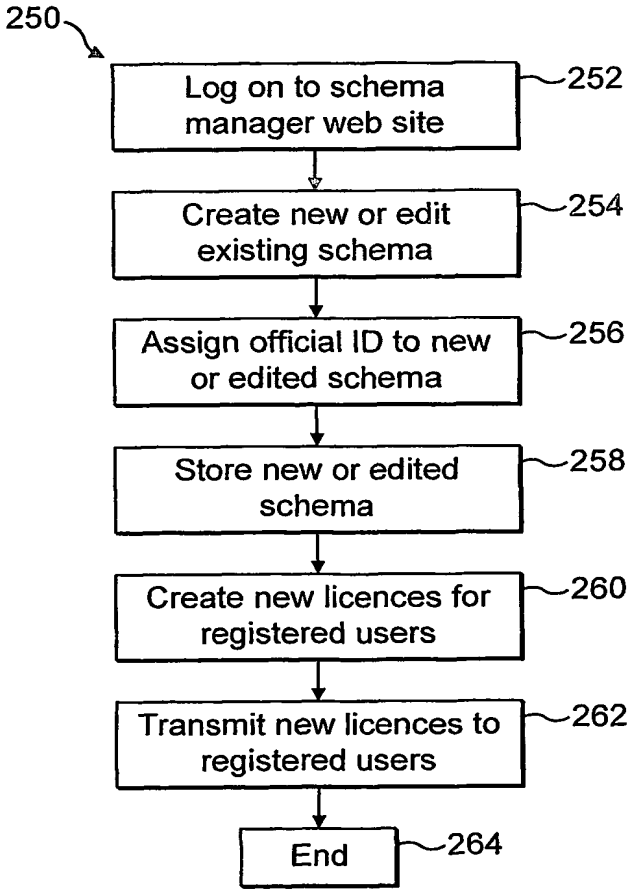
250
Log on to schema manager web site
252
Create new or edit existing schema
254
Assign official ID to new or edited schema
256
Store new or edited schema
258
Create new licences for registered users
260
Transmit new licences to registered users
262
End
264

FIG. 14

BEST AVAILABLE COPY

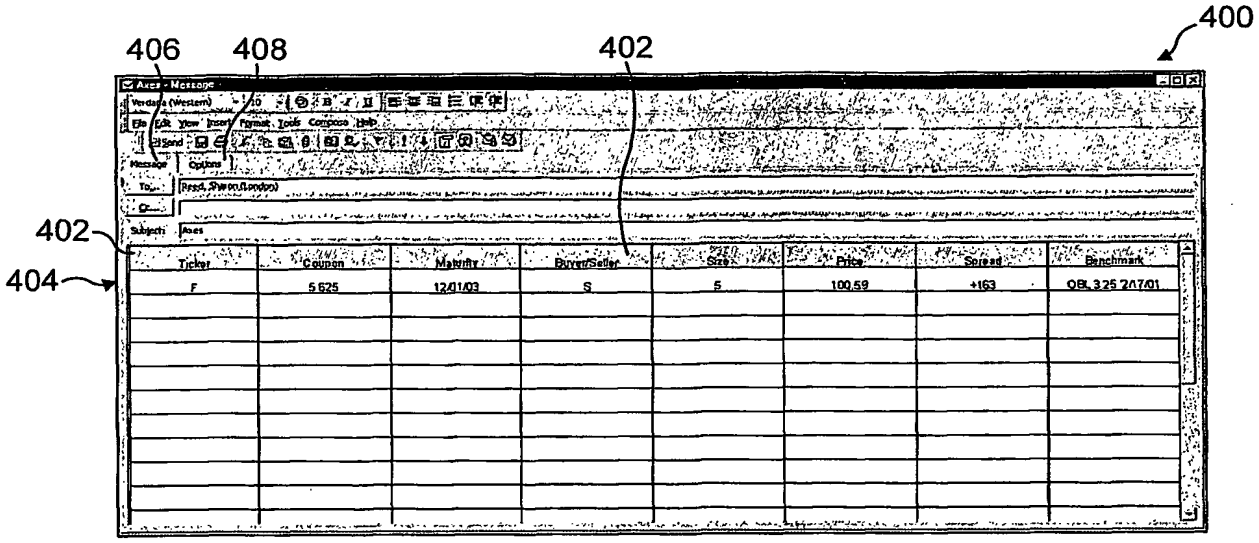

FIG. 15

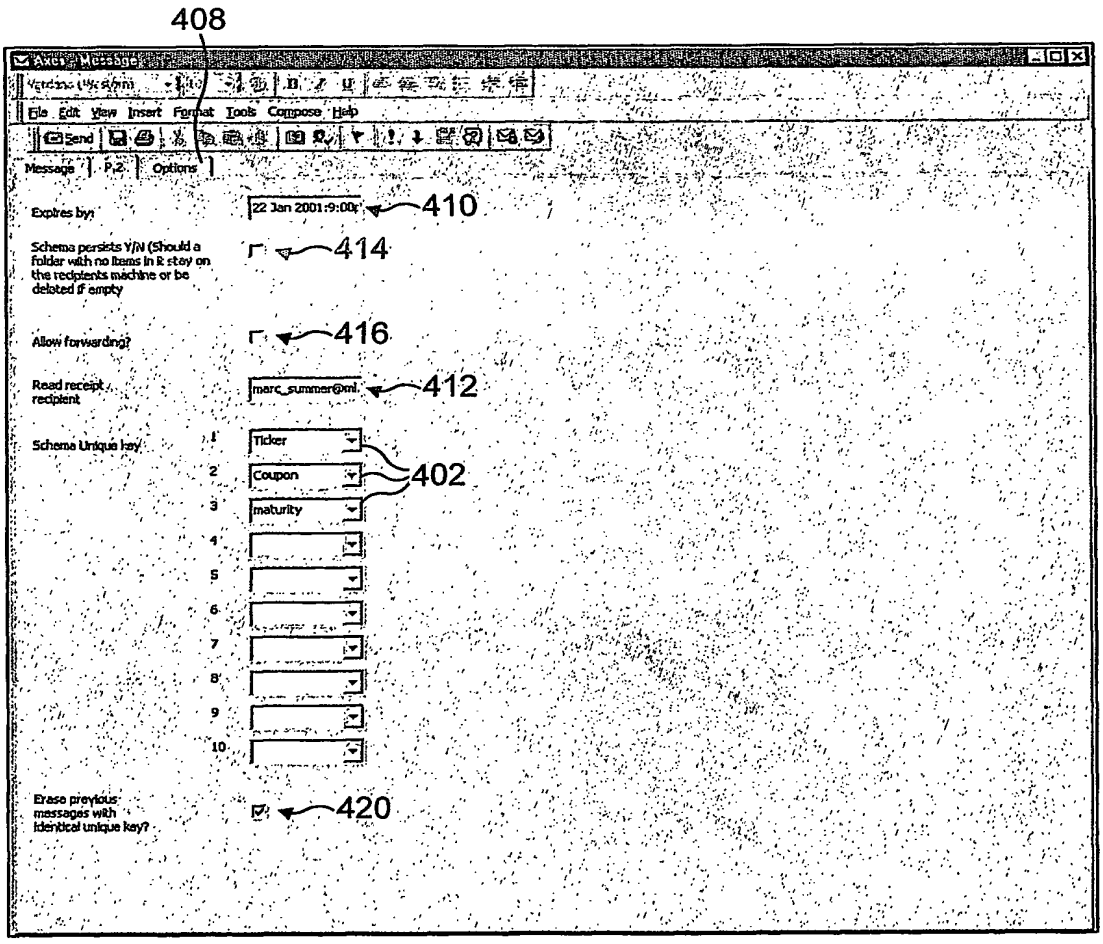

FIG. 16

BEST AVAILABLE COPY

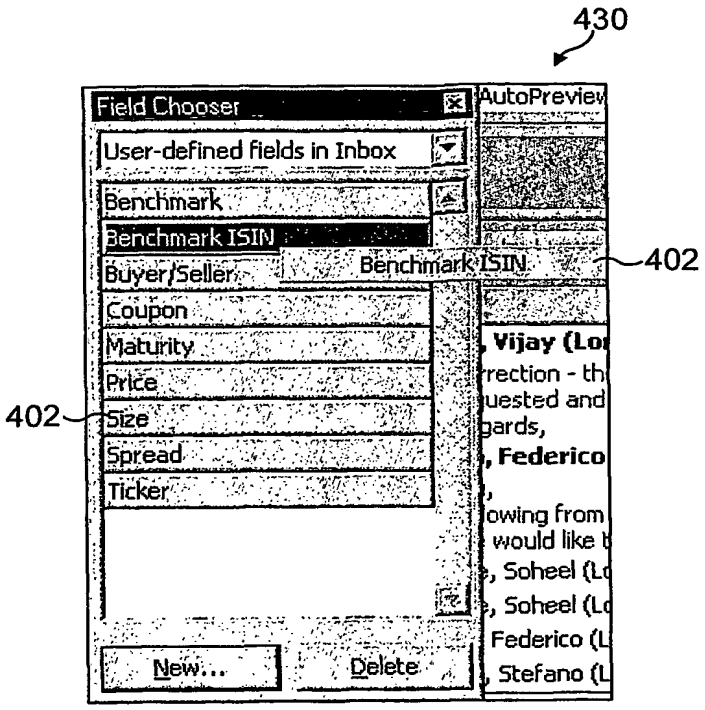

FIG. 17

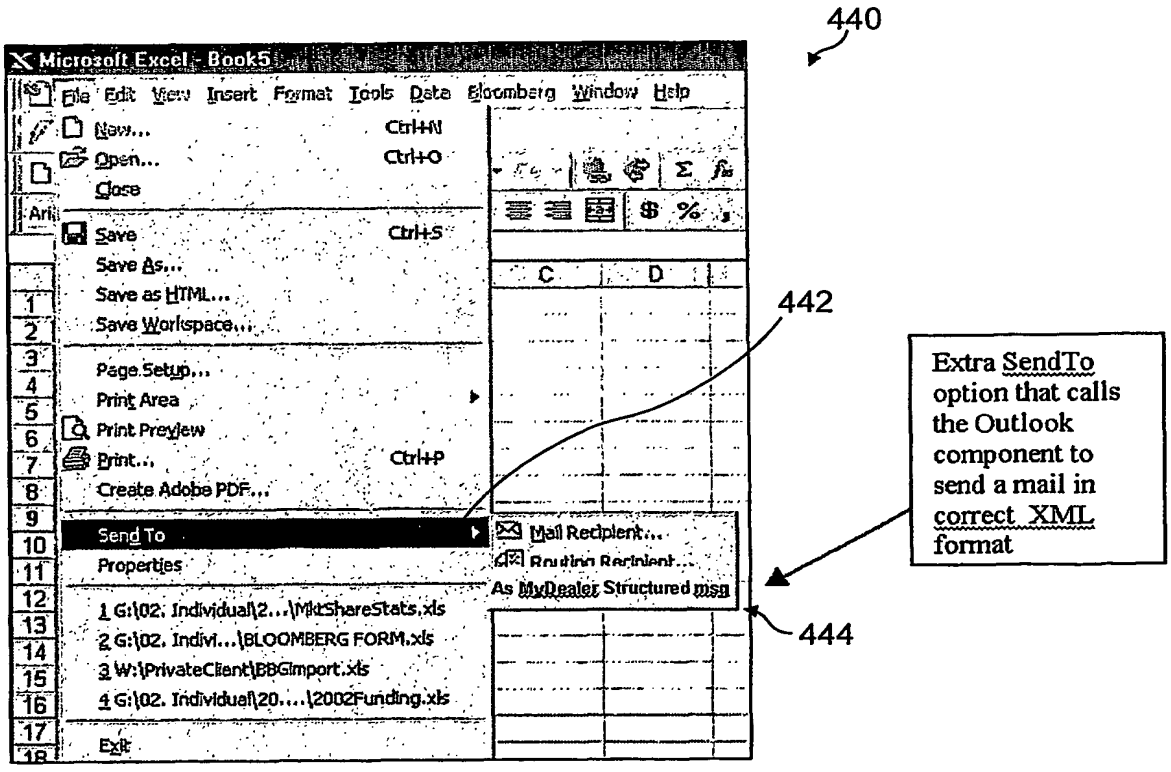

FIG. 18

Pseudo Code

```
..
CheckInBox
For Each Message
  If Message.Type = "Data"
    NewData = Decrypt (Message.Data)
    ParseData NewData
    Save NewData,NewData.DataName
  Else
    ..
  End if
  Else
    ..
  End if
Next
```

FIG. 19a

Department Manager PC / Filename = Salaries (original)

Sender,Name,Title,Age,Address,PostCode,Salary
HR(NY),J. Doe, Mr,33,1 Anywhere,10005,28000
HR (Boston),E. Kennedy,Chappaquidick Drive,12345,350000
HR(London),A. Smith,Ms,25,1 Armageddon,SN9 3AA,22000

FIG. 19b

New Data from sender "HR London":

Name	Title	Age	Address	PostCode	Salary
A. Smith	Ms	25	1 Armageddon	SN9 3AA	29000
J. Smith	Mr	45	45 Jones Drive	SN1 3AA	18000

FIG. 19c

E-mail

To: Department.Manager@client.com
From: HR.department.London@agency.com
Subject: Salary change and new hire

This e-mail contains data for your Steelhead software. If you can see this message, please goto www.steelhead.com/downloads to download the software.

```
<steelhead>
  <senderlicence>2h8d739487f9s8377h4hnnzx</senderlicence>
  <mailtype>data</mailtype>
  <uniquerecordID>Name,Title</uniquerecordID>
  <data>
    <dataname>Salaries</dataname>
    <row>
      <action>update</action>
      <Name>A. Smith</Name>
      <Title>Ms</Title>
      <Age>25</Age>
      <Address>1 Armageddon</Address>
      <PostCode>SN9 3AA</PostCode>
      <Salary>29000</PostCode>
    </row>
      <action>new</action>
      <Name>J. Smith</Name>
      <Title>Mr</Title>
      <Age>45</Age>
      <Address>45 Jones Drive</Address>
      <PostCode>SN1 3AA</PostCode>
      <Salary>18000</PostCode
  </data>
</steelhead>
```

FIG. 19d

Department Manager PC / Filename = Salaries (new)

Sender,Name,Title,Age,Address,PostCode,Salary
HR(NY),J. Doe, Mr,33,1 Anywhere,10005,28000
HR (Boston),E. Kennedy,Chappaquidick Drive,12345,350000
HR(London),A. Smith,Ms,25,1 Armageddon,SN9 3AA,29000
HR(London),J. Smith,Mr,45,45 Jones Drive,SN1 3AA,18000

FIG. 19e

Pseudo Code

```
..
LoadProgramFlowFile
..
CheckInBox
For Each Message
  If Message.Sender = "upgrades@steelhead.com"
    If Message.Subject = "Program-Flow Update"
      NewFileData = Decrypt (Message.Body)
      SaveProgramFlowFile (NewFileData)
      LoadProgramFlowFile
    Else
      ..
    End if
  Else
    ..
  End if
Next
```

FIG. 20a

Program-Flow File (original)

```
[menus]
Delete = True
Sort   = False
Filter = False

[authorisation]
ExpiryDate = 31/12/2002
CheckLicense = False
EnforceExpiry = False
AllowSending = False

[windows]
AllowMultipleWindows = False
```

FIG. 20b

E-mail

To: user@client.com
From: upgrades@steelhead.com
Subject: Program-Flow Update

```
[menus]
Delete = False
Archive = True
Sort   = True
Filter = True

[authorisation]
ExpiryDate = 31/12/2002
CheckLicense = True
EnforceExpiry = True
AllowSending = False

[windows]
AllowMultipleWindows = True
```

FIG. 20c

Program-Flow File (new)

```
[menus]
Delete = False
Archive = True
Sort   = True
Filter = True

[authorisation]
ExpiryDate = 31/12/2002
CheckLicense = True
EnforceExpiry = True
AllowSending = False

[windows]
AllowMultipleWindows = True
```

FIG. 20d

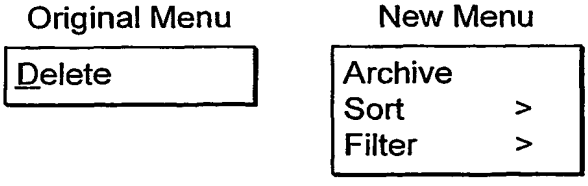

FIG. 20e

Pseudo Code

```
[Shell Program]
If FlagToLoadNewExeOnStartup
  Replace MainProgram with TemporaryExe
End If
Start MainProgram

[Main Program]
CheckInBox
For Each Message
  If Message.Sender = "upgrades@steelhead.com"
    If Message.Subject = "Full Update"
      NewFileData = Decrypt (Message.Body)
      Save NewFileData as TemporaryExe
      Set FlagToLoadNewExeOnStartup
    Else
      ..
    End if
  Else
    ..
  End if
Next
```

FIG. 21a

E-mail

To:	user@client.com
From:	upgrades@steelhead.com
Subject:	Full Update

Vndiofuynvlsvhn3987593e87nsk
Vnsdhjf577120n3945nbkshh93nb
Djeu348579ene9756bnseoodh93nr
Djndieurytnowiun39937970udnsk
.....

FIG. 21b

Pseudo Code

```
..
CheckInBox
For Each Message
  If Message.Sender = "commands@steelhead.com"
    CommandData = Decrypt (Message.Body)
    Get Command and Parameters from CommandData
    If Message.Subject = "Steelhead"
      Select Command
        LOGFILE:
           LogFileOutput (param1, param2))
        ERRORREPORT:
           Send E-mail ("errorreports@steelhead.com",
                        Errorreport(param1))
        BACKUPFILES:
        ..
      End Select
    Else
      ..
    End if
  Else
    ..
  End if
Next
```

FIG. 22a

E-mail

To: user@client.com
From: commands@steelhead.com
Subject: Steelhead

```
BACKUPFILES DATA,temp.dat
TESTDATA temp.dat,errors.log
```

FIG. 22b

E-mail

To: user@client.com
From: system@steelhead.com
Subject: Irrelevant Title

This e-mail contains harmless instructions to your Steelhead software. If you can see this message, please goto www.steelhead.com/downloads to download the software.

```
<steelhead>
  <senderlicence>837isn2855ns0d925n30</senderlicence>
  <mailtype>steelhead command</mailtype>
  <data>
    <command>
      <name>BACKUPFILES</name>
      <param1>DATA</param1>
      <param2>temp.dat</param2>
    </command>
    <command>
      <name>TESTDATA</name>
      <param1>temp.dat</param1>
      <param2>errors.log</param2>
    </command>
      </data>
</steelhead>
```

FIG. 22c

Pseudo Code

```
..
CheckInBox
For Each Message
  If Message.Sender = "commands@steelhead.com"
    CommandData = Decrypt (Message.Body)
    Get Command and Parameters from CommandData
    If Message.Subject = "Steelhead"
      ..
    Else
      Execute (Message.Subject) with Command and Parameters
      ..
    End if
  Else
    ..
  End if
Next
```

FIG. 23a

E-mail

To:	user@client.com
From:	commands@steelhead.com
Subject:	HomeSecurity.exe

HOME SECURITY, LIGHTS GARAGE,ON,NOW
ALARM ON,3416

FIG. 23b

E-mail

To:	user@client.com
From:	system@steelhead.com
Subject:	Instructions for HomeSecurity

If you can see this message, please goto www.steelhead.com/downloads to download the software.

```
<steelhead>
  <senderlicence>jf89fjds72938fs09983gnsk</senderlicence>
  <mailtype>external command</mailtype>
  <data>
    <program>
      <name>HomeSecurity.exe</name>
      <command>
        <name>LIGHTS</name>
        <param1 name="location">GARAGE</param1>
        <param2 name="state">ON</param2>
        <param3 name="time">NOW</param3>
      </command>
    </program>
  </data>
</steelhead>
```

FIG. 23c